

APPARATUS AND METHOD FOR A FLUSH
PROCEDURE IN AN INTERRUPTED TRACE STREAM

This application claims priority under 35 USC §119(e)(1) of Provisional Application Number 60/434,211 (TI-34659P) filed December 17, 2002.

Related Applications

- 5 U.S. Patent Application (Attorney Docket No. TI-34654),
entitled APPARATUS AND METHOD FOR SYNCHRONIZATION OF TRACE
STREAMS FROM MULTIPLE PROCESSORS, invented by Gary L.
Swoboda, filed on even date herewith, and assigned to the
assignee of the present application; U.S. Patent
10 Application (Attorney Docket No. TI-34655), entitled
APPARATUS AND METHOD FOR SEPARATING DETECTION AND ASSERTION
OF A TRIGGER EVENT, invented by Gary L. Swoboda, filed on

5 even date herewith, and assigned to the assignee of the
present application; U.S. Patent Application (Attorney
Docket No. TI- 34656), entitled APPARATUS AND METHOD FOR
STATE SELECTABLE TRACE STREAM GENERATION, invented by Gary
L. Swoboda, filed on even date herewith, and assigned to
10 the assignee of the present application; U.S. Patent
Application (Attorney Docket No. TI-34657), entitled
APPARATUS AND METHOD FOR SELECTING PROGRAM HALTS IN AN
UNPROTECTED PIPELINE AT NON-INTERRUPTIBLE POINTS IN CODE
EXECUTION, invented by Gary L. Swoboda and Krishna Allam,
15 filed on even date herewith, and assigned to the assignee
of the present application; U.S. Patent Application
(Attorney Docket No. TI-34658), entitled APPARATUS AND
METHOD FOR REPORTING PROGRAM HALTS IN AN UNPROTECTED
PIPELINE AT NON-INTERRUPTIBLE POINTS IN CODE EXECUTION,
20 invented by Gary L. Swoboda, filed on even date herewith,
and assigned to the assignee of the present application;
U.S. Patent Application (Attorney Docket No. TI-34660),
entitled APPARATUS AND METHOD FOR CAPTURING AN EVENT OR
COMBINATION OF EVENTS RESULTING IN A TRIGGER SIGNAL IN A
25 TARGET PROCESSOR, invented by Gary L. Swoboda, filed on
even date herewith, and assigned to the assignee of the
present application; U.S. Patent Application (Attorney
Docket No. TI-34661), entitled APPARATUS AND METHOD FOR
CAPTURING THE PROGRAM COUNTER ADDRESS ASSOCIATED WITH A
30 TRIGGER SIGNAL IN A TARGET PROCESSOR, invented by Gary L.
Swoboda, filed on even date herewith, and assigned to the

5 assignee of the present application; U.S. Patent
Application (Attorney Docket No. TI-34662), entitled
APPARATUS AND METHOD DETECTING ADDRESS CHARACTERISTICS FOR
USE WITH A TRIGGER GENERATION UNIT IN A TARGET PROCESSOR,
invented by Gary Swoboda and Jason L. Peck, filed on even
10 date herewith, and assigned to the assignee of the present
application; U.S. Patent Application (Attorney Docket No.
TI-34663), entitled APPARATUS AND METHOD FOR TRACE STREAM
IDENTIFICATION OF A PROCESSOR RESET, invented by Gary L.
Swoboda, Bryan Thome and Manisha Agarwala, filed on even
15 date herewith, and assigned to the assignee of the present
application; U.S. Patent (Attorney Docket No. TI-34664),
entitled APPARATUS AND METHOD FOR TRACE STREAM
IDENTIFICATION OF A PROCESSOR DEBUG HALT SIGNAL, invented
by Gary L. Swoboda, Bryan Thome, Lewis Nardini and Manisha
20 Agarwala, filed on even date herewith, and assigned to the
assignee of the present application; U.S. Patent
Application (Attorney Docket No. TI-34665), entitled
APPARATUS AND METHOD FOR TRACE STREAM IDENTIFICATION OF A
PIPELINE FLATTENER PRIMARY CODE FLUSH FOLLOWING INITIATION
25 OF AN INTERRUPT SERVICE ROUTINE; invented by Gary L.
Swoboda, Bryan Thome and Manisha Agarwala, filed on even
date herewith, and assigned to the assignee of the present
application; U.S. Patent Application (Attorney Docket No.
TI-34666), entitled APPARATUS AND METHOD FOR TRACE STREAM
30 IDENTIFICATION OF A PIPELINE FLATTENER SECONDARY CODE FLUSH
FOLLOWING A RETURN TO PRIMARY CODE EXECUTION, invented by

5 Gary L. Swoboda, Bryan Thome and Manisha Agarwala filed on
even date herewith, and assigned to the assignee of the
present application; U.S. Patent Application (Docket No.
TI-34667), entitled APPARATUS AND METHOD IDENTIFICATION OF
A PRIMARY CODE START SYNC POINT FOLLOWING A RETURN TO
10 PRIMARY CODE EXECUTION, invented by Gary L. Swoboda, Bryan
Thome and Manisha Agarwala, filed on even date herewith,
and assigned to the assignee of the present application; U.
S. Patent Application (Attorney Docket No. TI-34668),
entitled APPARATUS AND METHOD FOR IDENTIFICATION OF A NEW
15 SECONDARY CODE START POINT FOLLOWING A RETURN FROM A
SECONDARY CODE EXECUTION, invented by Gary L. Swoboda,
Bryan Thome and Manisha Agarwala, filed on even date
herewith, and assigned to the assignee of the present
application; U.S. Patent Application (Attorney Docket No.
20 TI-34669), entitled APPARATUS AND METHOD FOR TRACE STREAM
IDENTIFICATION OF A PAUSE POINT IN A CODE EXECUTION
SEQUENCE, invented by Gary L. Swoboda, Bryan Thome and
Manisha Agarwala, filed on even date herewith, and assigned
to the assignee of the present application; U.S. Patent
25 Application (Attorney Docket No. TI-34670), entitled
APPARATUS AND METHOD FOR COMPRESSION OF A TIMING TRACE
STREAM, invented by Gary L. Swoboda and Bryan Thome, filed
on even date herewith, and assigned to the assignee of the
present application; U.S. Patent Application (Attorney
30 Docket No. TI-34671), entitled APPARATUS AND METHOD FOR
TRACE STREAM IDENTIFICATION OF MULTIPLE TARGET PROCESSOR

5 EVENTS, invented by Gary L. Swoboda and Bryan Thome, filed
on even date herewith, and assigned to the assignee of the
present application; and U.S. Patent Application (Attorney
Docket No. TI-34672 entitled APPARATUS AND METHOD FOR OP
CODE EXTENSION IN PACKET GROUPS TRANSMITTED IN TRACE
10 STREAMS, invented by Gary L. Swoboda and Bryan Thome, filed
on even date herewith, and assigned to the assignee of the
present application are related applications.

15 **Background of the Invention**

1. Field of the Invention

 This invention relates generally to the testing of digital
20 signal processing units and, more particularly, to the
inclusion in the trace data streams of signals identifying
selected events in the digital signal processors under
test. These selected events are communicated to the
testing apparatus by signal groups. In transferring test
25 and debug data from the target processor to the host
processing unit, the data is arranged in packets. The
trace packets are formatted into export packets for the
actual transfer from the target processor.

5 2. Description of the Related Art

As microprocessors and digital signal processors have become increasingly complex, advanced techniques have been developed to test these devices. Dedicated apparatus is available to implement the advanced techniques. Referring to Fig. 1A, a general configuration for the test and debug of a target processor is shown. The test and debug procedures operate under control of a host processing unit 10. The host processing unit 10 applies control signals to the emulation unit 11 and receives (test) data signals from the emulation unit 11 by cable connector 14. The emulation unit 11 applies control signals to and receives (test) signals from the target processing unit 12 by connector cable 15. The emulation unit 11 can be thought of as an interface unit between the host processing unit 10 and the target processor 12. The emulation unit 11 must process the control signals from the host processor unit 10 and apply these signals to the target processor 12 in such a manner that the target processor will respond with the appropriate test signals. The test signals from the target processor 12 can be a variety types. Two of the most popular test signal types are the JTAG (Joint Test Action Group) signals and trace signals. The JTAG signal provides a standardized test procedure in wide use. Trace signals are signals from a multiplicity of junctions in the target processor 12. While the width of the bus interfacing to

5 the host processing unit **10** generally have a standardized width, the bus between the emulation unit **11** and the target processor **12** can be increased to accommodate the increasing complexity of the target processing unit **12**. Thus, part of the interface function between the host processing unit **10** and the target processor **12** is to store the test signals until the signals can be transmitted to the host processing unit **10**.

The data in trace stream that is transferred from the target processor to the host processing unit are originally formed in (10-bit) trace packets. In the example provided by Fig. 1B, the trace packets have two control bits and 8-bit payload. However, in the actual transmission from the target processor to the host processing unit, the 10-bit packets are arranged in 3-bit export packets, the 3-bit export trace packets being reassembled into 10-bit trace packets by the emulation unit or the host processing unit. Referring to Fig. 1C, the process of converting the 10-bit trace packets to the 3-bit trace packets is illustrated.

25 The 10-bit trace packets in the first column are converted to the 3-bit export trace packets of the second column. The third column indicates the number of bits that are carried forward from a 10-bit trace packet after each export trace packet is formed. The four 10-bit trace packets can contain a 32-bit word, i.e., in the 4 8-bit payloads. Also shown in Fig. 1C is the effect of halting

5 code execution in the target processor during a non-
interruptible code segment. The code will keep executing
until a suitable point has been reached. However, when the
last portion of the 32-bit word from the non-interruptible
code segment is finished, although the segment has been
10 completed, one bit of data remains to be transmitted.

A need has been felt for apparatus and an associated method
having the feature that data associated with an incomplete
export trace packet can be transmitted to a host processing
15 unit. It would be yet another feature of the apparatus and
associated method to transmit all of the data generated
during a non-interruptible code execution halt to the
testing apparatus. It would be yet another feature of the
apparatus and associated method to accommodate a non-
20 interruptible code halt in the export trace stream. It
would be a still further feature of the present invention
to provide export packets that fill a storage location.

5 **Summary of the Invention**

The aforementioned and other features are accomplished, according to the present invention, by providing a control signal that is activated when a non-interruptible code
10 execution receives a halt signal and the control signal is removed when the debug software directs the code to execute. The control signal is applied to the unit providing the export trace signals. When the control signal is active and when an incomplete export trace packet
15 is ready for transmission, the control signal causes the export trace packet to filled and transmitted to the testing apparatus. In addition, the presence of the control signal ensures that the export packet transmitted will contain as many data bits as would be present in the
20 export of a trace packet signal group, the trace packet signal group size being determined by the memory location in which the trace stream data will be stored.

Other features and advantages of present invention will be more clearly understood upon reading of the following
25 description and the accompanying drawings and the claims.

Brief Description of the Drawings

Figure 1A is a general block diagram of a system
30 configuration for test and debug of a target processor, while Fig. 1B illustrates the format of a trace packet; and

5 Fig. 1C illustrates the conversion from trace packets to export trace packets according to the prior art.

Figure 2 is a block diagram of selected components in the target processor used the testing of the central processing
10 unit of the target processor according to the present invention.

Figure 3 is a block diagram of selected components of the illustrating the relationship between the components
15 transmitting trace streams in the target processor.

Figure 4A illustrates format by which the timing packets are assembled according to the present invention, while Figure 4B illustrates the inclusion of a periodic sync ID
20 packet in the timing trace stream.

Figure 5 illustrates the parameters for sync markers in the program counter stream packets according to the present invention.

25 Figure 6A illustrates the sync markers in the program counter trace stream when a periodic sync point ID is generated, while Figure 6B illustrates the reconstruction of the target processor operation from the trace streams
30 according to the present invention.

5 Figure 7 is a block diagram illustrating the apparatus used in reconstructing the processor operation from the trace streams according to the present invention.

Figure 8 is block diagram of the apparatus for converting
10 group of trace packets to export packet when the trace packet is incomplete according to the present invention.

Figure 9 illustrates the conversion of the trace packets to export packets according to the present invention.

15

Description of the Preferred Embodiment

1. Detailed Description of the Figures

20 Fig. 1A, Fig. 1B and Fig. 1C been described with respect to the related art.

Referring to Fig. 2, a block diagram of selected components of a target processor **20**, according to the present
25 invention, is shown. The target processor includes at least one central processing unit **200** and a memory unit **208**. The central processing unit **200** and the memory unit **208** are the components being tested. The trace system for testing the central processing unit **200** and the memory unit
30 **202** includes three packet generating units, a data packet generation unit **201**, a program counter packet generation

5 unit **202** and a timing packet generation unit **203**. The data
packet generation unit **201** receives VALID signals,
READ/WRITE signals and DATA signals from the central
processing unit **200**. After placing the signals in packets,
the packets are applied to the scheduler/multiplexer unit
10 **204** and forwarded to the test and debug port **205** for
transfer to the emulation unit **11**. The program counter
packet generation unit **202** receives PROGRAM COUNTER
signals, VALID signals, BRANCH signals, and BRANCH TYPE
signals from the central processing unit **200** and, after
15 forming these signal into packets, applies the resulting
program counter packets to the scheduler/multiplexer **204**
for transfer to the test and debug port **205**. The timing
packet generation unit **203** receives ADVANCE signals, VALID
signals and CLOCK signals from the central processing unit
20 **200** and, after forming these signal into packets, applies
the resulting packets to the scheduler/multiplexer unit **204**
and the scheduler/multiplexer **204** applies the packets to
the test and debug port **205**. Trigger unit **209** receives
EVENT signals from the central processing unit **200** and
25 signals that are applied to the data trace generation unit
201, the program counter trace generation unit **202**, and the
timing trace generation unit **203**. The trigger unit **209**
applies TRIGGER and CONTROL signals to the central
processing unit **200** and applies CONTROL (i.e., STOP and
30 START) signals to the data trace generation unit **201**, the
program counter generation unit **202**, and the timing trace

5 generation unit **203**. The sync ID generation unit **207**
applies signals to the data trace generation unit **201**, the
program counter trace generation unit **202** and the timing
trace generation unit **203**. While the test and debug
apparatus components are shown as being separate from the
10 central processing unit **201**, it will be clear that an
implementation these components can be integrated with the
components of the central processing unit **201**.

Referring to Fig. 3, the relationship between selected
15 components in the target processor **20** is illustrated. The
data trace generation unit **201** includes a packet assembly
unit **2011** and a FIFO (first in/first out) storage unit
2012, the program counter trace generation unit **202**
includes a packet assembly unit **2021** and a FIFO storage
20 unit **2022**, and the timing trace generation unit **203**
includes a packet generation unit **2031** and a FIFO storage
unit **2032**. As the signals are applied to the packet
generators **201**, **202**, and **203**, the signals are assembled
into packets of information. The packets in the preferred
25 embodiment are 10 bits in width. Packets are assembled in
the packet assembly units in response to input signals and
transferred to the associated FIFO unit. The
scheduler/multiplexer **204** generates a signal to a selected
trace generation unit and the contents of the associated
30 FIFO storage unit are transferred to the
scheduler/multiplexer **204** for transfer to the emulation

5 unit. Also illustrated in Fig. 3 is the sync ID generation unit **207**. The sync ID generation unit **207** applies an SYNC ID signal to the packet assembly unit of each trace generation unit. The periodic signal, a counter signal in the preferred embodiment, is included in a current packet
10 and transferred to the associated FIFO unit. The packet resulting from the SYNC ID signal in each trace is transferred to the emulation unit and then to the host processing unit. In the host processing unit, the same count in each trace stream indicates that the point at
15 which the trace streams are synchronized. In addition, the packet assembly unit **2031** of the timing trace generation unit **203** applies an INDEX signal to the packet assembly unit **2021** of the program counter trace generation unit **202**. The function of the INDEX signal will be described below.

20

Referring to Fig. 4A, the assembly of timing packets is illustrated. The signals applied to the timing trace generation unit **203** are the CLOCK signals and the ADVANCE signals. The CLOCK signals are system clock signals to
25 which the operation of the central processing unit **200** is synchronized. The ADVANCE signals indicate an activity such as a pipeline advance or program counter advance (()) or a pipeline non-advance or program counter non-advance (1). An ADVANCE or NON-ADVANCE signal occurs each clock
30 cycle. The timing packet is assembled so that the logic signal indicating ADVANCE or NON-ADVANCE is transmitted at

5 the position of the concurrent CLOCK signal. These combined CLOCK/ADVANCE signals are divided into groups of 8 signals, assembled with two control bits in the packet assembly unit **2031**, and transferred to the FIFO storage unit **2032**.

10

Referring to Fig. 4B, the trace stream generated by the timing trace generation unit **203** is illustrated. The first (in time) trace packet is generated as before. During the assembly of the second trace packet, a SYYN ID signal is
15 generated during the third clock cycle. In response, the timing packet assembly unit **2031** assembles a packet in response to the SYNC ID signal that includes the sync ID number. The next timing packet is only partially assembled at the time of the SYNC ID signal. In fact, the SYNC ID
20 signal occurs during the third clock cycle of the formation of this timing packet. The timing packet assembly unit **2031** generates a TIMING INDEX 3 signal (for the third packet clock cycle at which the SYNC ID signal occurs) and transmits this TIMING INDEX 3 signal to the program counter
25 packet assembly unit **2031**.

Referring to Fig. 5, the parameters of a sync marker in the program counter trace stream, according to the present invention is shown. The program counter stream sync
30 markers each have a plurality of packets associated therewith. The packets of each sync marker can transmit a

5 plurality of parameters. A SYNC POINT TYPE parameter defines the event described by the contents of the accompanying packets. A program counter TYPE FAMILY parameter provides a context for the SYNC POINT TYPE parameter and is described by the first two most
10 significant bits of a second header packet. A BRANCH INDEX parameter in all but the final SYNC POINT points to a bit within the next relative branch packet following the SYNC POINT. When the program counter trace stream is disabled, this index points a bit in the previous relative branch
15 packet when the BRANCH INDEX parameter is not a logic "0". In this situation, the branch register will not be complete and will be considered as flushed. When the BRANCH INDEX is a logic "0", this value point to the least significant value of branch register and is the oldest branch in the
20 packet. A SYNC ID parameter matches the SYNC POINT with the corresponding TIMING and/or DATA SYNC POINT which are tagged with the same SYNC ID parameter. A TIMING INDEX parameter is applied relative to a corresponding TIMING SYNC POINT. For all but LAST POINT SYNC events, the first
25 timing packet after the TIMING PACKET contains timing bits during which the SYNC POINT occurred. When the timing stream is disabled, the TIMING INDEX points to a bit in the timing packet just previous to the TIMING SYNC POINT packet when the TIMING INDEX value is nor zero. In this
30 situation, the timing packet is considered as flushed. A TYPE DATA parameter is defined by each SYNC TYPE. An

5 ABSOLUTE PC VALUE is the program counter address at which the program counter trace stream and the timing information are aligned. An OFFSET COUNT parameter is the program counter offset counter at which the program counter and the timing information are aligned.

10

Referring to Fig. 6A, a program counter trace stream for a hypothetical program execution is illustrated. In this program example, the execution proceeds without interruption from external events. The program counter

15 trace stream will consist of a first periodic sync point marker 601, a plurality of periodic sync point ID markers 602, and last sync point marker 603 designating the end of the test procedure. The principal parameters of each of the packets are a sync point type, a sync point ID, a

20 timing index, and an absolute PC value. The first and last sync points identify the beginning and the end of the trace stream. The sync ID parameter is the value from the value from the most recent sync point ID generator unit. In the preferred embodiment, this value is a 3-bit logic sequence.

25 The timing index identifies the status of the clock signals in a packet, i.e., the position in the 8 position timing packet when the event producing the sync signal occurs. And the absolute address of the program counter at the time that the event causing the sync packet is provided. Based

30 on this information, the events in the target processor can be reconstructed by the host processor.

5

Referring to Fig. 6B, the reconstruction of the program execution from the timing and program counter trace streams is illustrated. The timing trace stream consists of packets of 8 logic "0"s and logic "1"s. The logic "0"s indicate that either the program counter or the pipeline is advanced, while the logic "1"s indicate the either the program counter or the pipeline is stalled during that clock cycle. Because each program counter trace packet has an absolute address parameter, a sync ID, and the timing index in addition to the packet identifying parameter, the program counter addresses can be identified with a particular clock cycle. Similarly, the periodic sync points can be specifically identified with a clock cycle in the timing trace stream. In this illustration, the timing trace stream and the sync ID generating unit are in operation when the program counter trace stream is initiated. The periodic sync point is illustrative of the plurality of periodic sync points that would typically be available between the first and the last trace point, the periodic sync points permitting the synchronization of the three trace streams for a processing unit.

Referring to Fig. 7, the general technique for reconstruction of the trace streams is illustrated. The trace streams originate in the target processor **12** as the target processor **12** is executing a program **1201**. The trace

5 signals are applied to the host processing unit **10**. The
host processing unit **10** also includes the same program
1201. Therefore, in the illustrative example of Fig. 6
wherein the program execution proceeds without
interruptions or changes, only the first and the final
10 absolute addresses of the program counter are needed.
Using the advance/non-advance signals of the timing trace
stream, the host processing unit can reconstruct the
program as a function of clock cycle. Therefore, without
the sync ID packets, only the first and last sync markers
15 are needed for the trace stream. This technique results in
reduced information transfer. Fig. 6 includes the presence
of periodic sync ID cycles, of which only one is shown.
The periodic sync ID packets are important for
synchronizing the plurality of trace streams, for selection
20 of a particular portion of the program to analyze, and for
restarting a program execution analysis for a situation
wherein at least a portion of the data in the trace data
stream is lost. The host processor can discard the
(incomplete) trace data information between two sync ID
25 packets and proceed with the analysis of the program
outside of the sync timing packets defining the lost data.

Referring to Fig. 8, the apparatus for converting the trace
packets to export trace packets is shown. Trace packets
30 are applied to trace packets unit **81**. In trace packets
units **81**, the trace packets are grouped into export trace

5 packets. When an export trace packet is available, a
PACKET AVAILABLE signal is applied to logic "OR" gate **83**
and to a control terminal of multiplexer **82**. When the
PACKET AVAILABLE signal is present, an export race packet
is transmitted through the multiplexer **82** and applied to
10 export trace unit **84**. From export trace unit **84**, the
export trace packets are transferred to the host processing
unit (not shown). When an export trace packet is received
by the export trace unit **84**, a PACKET ACKNOWLEDGE signal is
applied to the trace packets unit **81** and the flush packet
15 unit **86**. 1-bit register **85** receives a HALT DURING A
NON_INTERRUPTIBLE CODE SEGMENT signal. This signal sets a
bit in 1-bit register **85**. The bit in 1-bit register **85**
applies a control signal to flush packet unit **86**. The flush
packet unit **86** has trace packets applied thereto and
20 applies a PACKET AVAILABLE signal to a second input
terminal of logic "OR" gate **83**. In the presence of the
control signal applied to flush packet unit **86**, flush
packets are applied through the multiplexer **82** to the
export trace packet unit **84**.

25

Referring to Fig. 9, the operation of the present invention
is illustrated. The trace packets are generated and
converted to export trace packets. After a halt is
signaled during a non-interruptible code segment, execution
30 of the code segment is continued until an appropriate halt
point is found. The present invention, as shown in Fig. 9,

5 stops generating trace packets. Export trace packets are
generated for the trace packets that have been generated.
However, when there is a remainder, the flush trace unit
generates a flush packet that completes the data generated
by the non-interruptible code segment. The flush packet
10 will add logic "0"s to the incomplete packets. In
addition, flush packets will be generated to provide
sufficient logic signals to populate a standard memory
location in the memory unit. When the target processor
begins operation after a pause, the trace packets are
15 converted into export trace packets as before.

5 2. Operation of the Preferred Embodiment

The present invention provides a technique for completing the transfer of trace data after the generation of halt signal for a non-interruptible code segment. Because of
10 the nature of the code segment, the data must be transmitted as developed and not be retained in the target processor. Because the export trace packets are a different length as compared to the trace packets, the last export trace packet may not be fully populated. In the
15 event that a remainder is present, a flush packet is generated to transfer the incomplete packet to the host processing unit. Logic "0"s complete the contents of the flush packets. In addition, flush packets are generated to insure that logic signals are available to populate the
20 memory locations into which the packet group payloads are being entered.

The present invention relies on the ability of relate the timing trace stream and the program counter trace stream.
25 This relationship is provided by having periodic sync ID information transmitted in each trace stream. In addition, the timing packets are grouped in packets of eight signals identifying whether the program counter or the pipeline advanced or didn't advance. The sync markers in the
30 program counter stream include both the periodic sync ID and the position in the current eight position packet when

5 the event occurred. Thus, the clock cycle of the event can
be specified. In addition, the address of the program
counter is provided in the program counter sync markers so
that the debug halt event can be related to the execution
of the program.

10

While the invention has been described with respect to the
embodiments set forth above, the invention is not
necessarily limited to these embodiments. Accordingly,
15 other embodiments, variations, and improvements not
described herein are not necessarily excluded from the
scope of the invention, the scope of the invention being
defined by the following claims.